Carnegie Mellon

Software Architecture for Managers

The Business Context

Instructor: Nguyễn Đức Mận
0904 235 945
mannd@duytan.edu.vn

Nội dung được xây dựng dựa trên tài liệu của Prof. Matthew Bass 2018 - ISR

1

Session Outline

- What is the business context?
- Organizational categories
- Summary

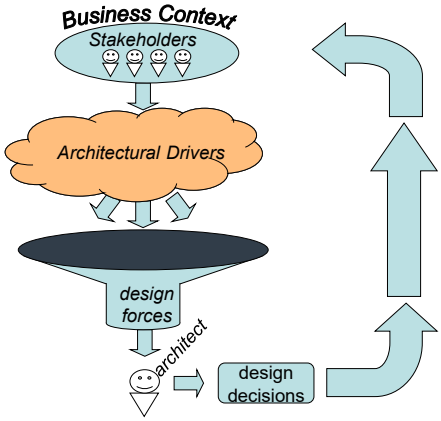
2

Session Outline

- What is the business context?
- Organizational categories
- Summary

3

Recall …



4

But What Is The "Business Context"?

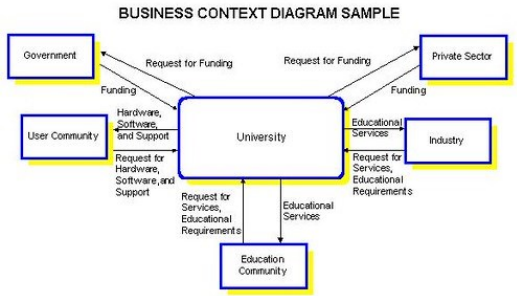
- The design should support the needs of the business context
- But what is the business context?
- And which portion of the business context is important?

Architectural Concerns

- Remember the architectural drivers
  - Functional requirements
  - Quality attribute requirements
  - Business constraints
  - Technical constraints
- The "Business Context" gives rise to these concerns
- The context also places constraints on how we realize the architecture

Business Context

- The Architectural Drivers typically come from:
  - Business goals/strategy
  - Organizational context
  - Domain related concerns
  - Regulatory concerns
  - Existing technical environment



7

Business Goals

- Organizations all have objectives
  - For commercial organizations these are typically motivated by revenue incentives
- Achieving these objectives is somehow related to the software under consideration
  - Knowing how they are related will help us to better understand (and predict) the drivers
  - In other words the strategy that the organization formulated includes the development of a software intensive system

8

Organizational Context

- In addition to the strategy we need to understand the organizational context
- This is going to impose implicit constraints on the system
- For example:
  - Who will develop the system
  - Who will maintain the system
  - Is the release dependent on other parts of the organization
  - Is the release dependent on other software or hardware co-development?
  - What's the impact if we "get something wrong"?
  - …

Organizational Impact

- These concerns have an impact on the structure of the system
  - Although the impact isn't always obvious
- For example, knowing who will develop the system can tell you:
  - What knowledge and skills they have
  - How easily they can coordinate
  - What their work load is like
  - …
- Additionally, these constrain how we realize the architecture
  - We'll see more about this in a minute

Domain Related Concerns

- The domain itself often dictates concerns
  - Consider consumer electronics
- A large percentage of the sales occur during the Xmas sales cycle
  - Missing the sales cycle will reduce the value of the system to the organization
- Perhaps the domain is rapidly evolving
  - If the system doesn't support rapid evolution the organization will be left behind
- The domain may requires regulatory compliance

Regulations

- Many organizations are regulated e.g.
  - Electronic health related data
  - Accounting practices for publically held companies
  - Transportation industry
  - Pharmaceutical manufacturing
  - Building automation
  - …

Regulatory Impact

- These regulations can imply:
  - Particular processes
  - Functional requirements
  - Quality attribute requirements
  - Limit technology choices
- Regulations often change regularly
  - The systems need to adapt to changing regulations

Organizational Characteristics

- Organizations of a similar type have consistent characteristics
- Knowing something about these characteristics will given insight into where the drivers will come from
  - This knowledge will also inform the lifecycle practices that will be more likely to be effective
- Let's look at some examples

Session Outline

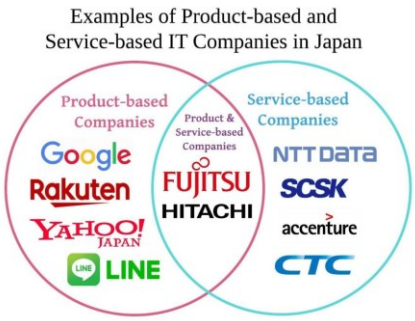
- What is the business context?
- Organizational categories
- Summary

"Categories" of Companies

- Product oriented companies
- Enterprises
- "Service based companies"

"Categories" of Companies

- Product oriented companies
- Enterprises
- "Service based companies"



Product Oriented Companies

- These organizations develop products that have a significant software component
  - These could be software intensive products (e.g. cell phones, automation systems, …)
  - These could be shrink wrapped software (e.g. Microsoft Office, Adobe Acrobat, …)
- The design and development of these products is a significant investment

Characterization of a “Product”

- For the purposes of this discussion a “product” has a dependent distribution channel
- That means that the development team doesn’t deploy the product directly
  - For example in the cloud
- This means the development team needs to coordinate distribution

Distribution Model

- The product can be manufactured, packaged, and shipped such as a:
  - Cell phone
  - TV
  - Ultrasound machine
- Or it could be distributed either on a disc or downloaded by the consumer e.g.
  - Adobe Acrobat
  - Microsoft Office

Preparation of the Sales Channel

- In order for people/organizations to buy the product they need to know about it
- There is often some promotional activity that accompanies a release
- The release is coordinated with the release of other products e.g.
  - If the old product is still selling well perhaps the release of the new product will be delayed

Market Analysis

- The products are conceived to support an identified market need
- Market characteristics include:
  - Market size
  - Needs of the market
  - Competitive products
  - Viable product price
- Organizations often spend time analyzing the market to inform the specification of the product
- The analysis is reflected in a “product roadmap”

Product Roadmap

- The roadmap will specify the products to be developed over some period of time
- This roadmap often specifies
  - Intended market
  - Market intent (position of the product in the market)
  - Anticipated number of units that will sell
  - Features/capability
  - Target margin per unit

Product Strategy

- Organizations often build products from a “product line”
- A product line is a core set of assets designed to support an identified family of similar products
- This means that the system (or “core assets”) needs to support the range of product needs

Product Features

- The products might vary in terms of:
  - BOM costs (so less memory for example)
  - Support for rear seat entertainment
  - Support for 3D graphics
  - Regulatory concerns (e.g. rear camera start up time)
- But is consistent in many ways as well

If You Get It Wrong?

- The impact of getting something wrong can be disastrous e.g.:
  - Wrong product for the market
  - Poor quality
  - Doesn't support required quality attributes
  - Isn't usable by target market
  - …
- Any of these can:
  - Adversely impact reputation
  - Reduce current and future sales
  - Be difficult to fix in a timely manner

Increased Upfront Effort

- This all results in an increased upfront effort
  - Meaning larger capital investment
- The organization needs to realize a return on investment
- Success depends on things like:
  - The development effort required to realize the products on the roadmap
  - The characteristics of each product
  - The costs to maintain the products

Business Context

- For product oriented organizations the context is going to be dominated by the product roadmap
- The “quality attributes” are often not explicitly identified in the roadmap
  - Although they are required
  - When they are, they are typically isolated to the attributes of a given product
- You also need to look across products to determine the characteristics of the “core asset” or platform

Discussion in group

- **Why a Focused Project Context Is Necessary?**
- **Ex:**
  - Stephen Johnson is a photographer wishing to build a growth strategy for his newly opened shop. Stephen's project definition states a goal of "seeking to grow *Johnson Photography* by creating more value for *customers* than *Frederick Pictures* (the local competitor)."
  - Who is CUSTOMERS? ARE THEY DIFFERENT?

Refining the Roadmap

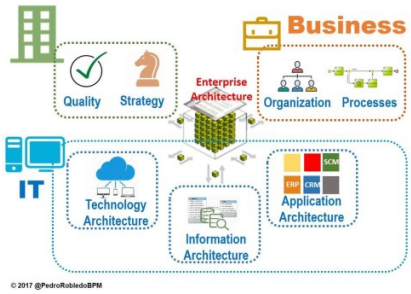
- The initial roadmap might impose demands on the system that are not easily achievable
- If that's the case the organization needs to recognize this
  - They need to know the nature of the limitations
  - Only then will they be in a position to refine the roadmap

Several "Categories" of Companies

- Product oriented companies
- Enterprises
- "Service based companies"

Enterprise Organizations

- Do not develop software as the primary business
- Business is enabled by software
  - Could be back office or customer facing
- Examples include:
  - Banking
  - Finance
  - Oil and Gas
  - Insurance
  - …



Characteristics

- Dependent on software but it isn't their primary business
- Have grown organically and through acquisition
- Have a large proliferation of disparate systems
- Some of their systems automate the business processes within an organization
  - Others are customer facing applications
- Typically are interested in enabling "business agility"
  - Will often see the goal articulated as "reducing complexity"

Dynamic Organizational Environment

- These organizations are dynamic in many respects
- They are regularly reorganizing due to:
  - Acquisitions
  - Mergers
  - Internal reorganization
- They are often regulated in one way or another
  - These regulations change all the time

Inhibited by IT

- These organizations are often inhibited by IT
- The effort required to reorganize is impacted by the IT systems
- This affects
  - Agility of the organization
  - Operational costs
  - Ability to respond to the needs of the customer
  - Reputation of the organization

Strategic Objectives

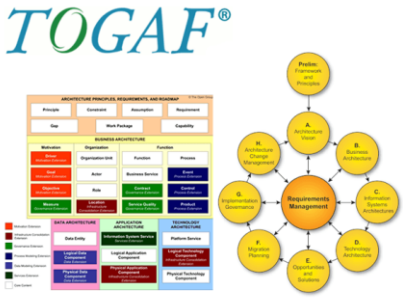
- In one way or another the system need to support the strategic objectives of the organization
- This includes things like:
  - Operational costs
  - Ability to pursue specific objectives
  - Ability to pursue objectives that are yet to be identified
  - Business continuity
  - Growth objectives
  - Ability to analyze business
- These concerns give rise to both functionality and systemic concerns (quality attributes)

Enterprise Architecture

- Many enterprises are focused on what's often called "Enterprise Architecture" related activities
- This means they are thinking about how to align the structure of their enterprise with the goals
- Often the drivers for the *software architecture* are in part related to this activity

Enterprise Architecture Frameworks

- There are several frameworks that organizations use to structure their EA activities
  - TOGAF
  - Zachman Framework
  - DODAF
  - …

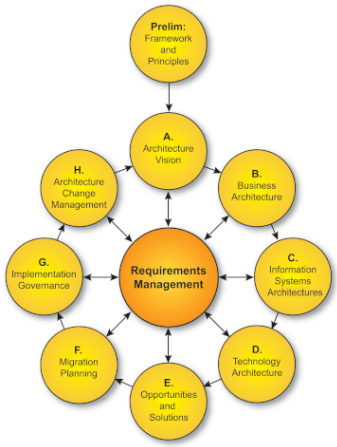


TOGAF

- TOGAF (and similar frameworks) are focused on the *Enterprise*
  - This means the elements of the organization, the relationships among them, and the business processes
- TOGAF does use a lot of words that sound similar to what we say
  - It is not, however, explicitly focused on the software architecture

Enterprise Architecture Development Method

- ADM is a part of TOGAF
- It describes how you design and evolve the EA



Phases of ADM

- Architecture Vision
- Business Architecture
- Information Systems Architecture
- Technology Architecture
- Opportunities and Solutions
- Migration Planning
- Implementation Governance
- Architecture Change Management

Software Architecture?

- The language is confusing
  - TOGAF talks about architecture
  - It talks about Information and Technology architecture
- The meaning, however, is different
  - If you look at the details you realize that TOGAF is not referring to the same concerns we have been talking about
- There is a relationship, however

Drivers?

- The business goals often imply architectural drivers although they are often implicit e.g.
  - Reduce complexity is the number one reported goal among CIOs
- The reason for this goal is to reduce the effort to maintain the system
  - In other words reduce the cost of change

Translating

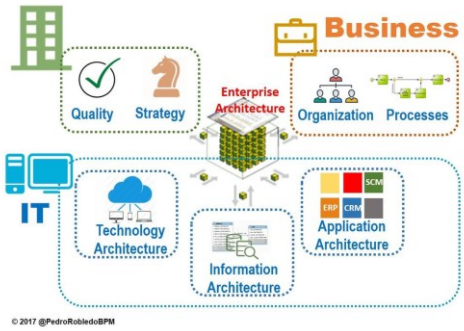
- Understanding where to look for business drivers is important
- The typical thing is to pay attention to the business processes
- The business goals aren't typically considered, however
- You need to know to consider:
  - How many?
  - How often?
  - How fast?
  - …

Roll-Out

- As these systems automate the business processes, they impact the way people work
- These processes typically span multiple portions of the organization
  - Likely different management chains
- As a result roll-out needs to be coordinated
  - This is something that takes time and effort
- Again, the impact of getting it wrong can be bad
  - Slow down or disrupt business
  - Impact adoption of solution
  - Reduce efficiency
  - …

Several “Categories” of Companies

- Product oriented companies
- Enterprises
- “Service based companies”



Service Based Companies

- The name is a misnomer (but I don't have a better one)
- This describes organizations that deploy their software predominantly in the cloud (private or public) and offer it as a service
- Examples organizations are:
  - Amazon
  - Google
  - Netflix
  - Turn
  - Groupon
  - Linkedin
  - …

Characteristics

- Continuous deployment lifecycle
- Flat organizations
- Empowered developers
- Large scale applications (in terms of data and/or users)
- Rapidly emerging and changing businesses

Continuous Deployment

- The development team releases the software directly
- They don't need to inform anyone or coordinate with external teams
- They release often
  - Typically not longer than once a week

Dynamic Environment

- Due to the nature of the systems they are difficult to test fully prior to release
  - These are very large scale distributed systems that run in an environment that's constantly changing
- These organizations typically use "live testing"
  - This means they test the system while in production

Flat Organization

- In these organizations the development team identifies the specific requirements
- They prioritize the requirements according to their understanding of the needs of their customer
- They are not evaluated based on schedule and budget
  - Again, they don't need to notify anyone of a release

Customer Centric Development

- This doesn't mean they don't have systemic concerns
- It means that they need to understand their customer enough to determine what these concerns are
- They often view historic usage patterns to gain insight into customer's needs
  - Product managers can help with this

Evaluate Architecture

- The system overall is monitored and evaluated continuously
- If desired systemic properties are inhibited the system changes can be rolled back
- User behavior can also be monitored in real time
  - Some times multiple alternatives will be provided

Session Outline

- What is the business context?
- Organizational categories
- Summary

Summary

- The system needs to support the business context
  - This is true regardless of the context
- Different kinds of organizations imply very different kinds of business contexts
  - Knowing something about these patterns can give you insight into the nature of the context
- The context will also dictate the way in which concerns can be elicited and validated
  - And in fact dictates often the lifecycle model as well

.References

- Bass, L.; Clements, P. & Kazman, R. *Software Architecture in Practice, Second Edition.* Boston, MA: Addison-Wesley, 2003.
- Shang, S; Seddon, P. *Assessing and managing the benefits of enterprise systems: the business manager's perspective* Information System Journal, issue 12 p271-299, 2002
- Video for Context Diagram https://youtu.be/fWNrc6GNK14